

Полная программа

JavaScript-разработчик. Продвинутый уровень

Научитесь подбирать и использовать полный ландшафт современных технологий Node.js / React / Vue / TypeScript / Web Components

Длительность курса: 130 часов

Модуль 1. Продвинутые основы JavaScript и TypeScript

Тема 1

Введение в курс: знакомство с программой, командой курса

Цель занятия

познакомиться с преподавателем и с программой курса; вспомнить основы JS.

Краткое содержание

организационные вопросы по курсу; типы данных; принципы работы с JS кодом.

Тема 2

ООП в JavaScript

Цель занятия

познакомиться с основами объектно-ориентированного программирования в JavaScript.

Краткое содержание

классы и наследование; прототипы; инкапсуляция и полиморфизм.

Домашние задания

Глубокое сравнение объектов (deepEqual)

Цель

научиться применять ООП-подходы (классы, наследование, инкапсуляция, полиморфизм) для решения реальных задач на JavaScript и закрепить понимание прототипов и работы с объектами, а также реализовать полезную утилиту для сравнения сложных структур данных.

Тема 3

Основы функциональной разработки

Цель занятия

освоить базовые концепции функционального программирования в JavaScript.

Краткое содержание

функции высшего порядка; чистые функции; неизменяемость.

Тема 4

Введение в TypeScript: базовые типы, интерфейсы, функции

Цель занятия

изучить основы типизации в TypeScript; освоить типы, интерфейсы и функции.

Краткое содержание

базовые типы данных TypeScript; интерфейсы и типы; типизация функций и объектов.

Домашние задания

Глубокое сравнение объектов (deepEqual) на TypeScript

Цель

научиться применять строгую типизацию и инструменты TypeScript для реализации сложных функций; закрепить навыки работы с типами, интерфейсами и аннотациями типов в реальных задачах.

Модуль 2. Инфраструктура фронтенда и базовые подходы

Тема 1

Управление состоянием приложения, Redux и работа с actions (исторический контекст)

Цель занятия

освоить Redux Toolkit.

Краткое содержание

Redux Toolkit; состояние; actions; reducers.

Тема 2

Клиентский роутинг, построение SPA

Цель занятия

изучить роутинг и SPA.

Краткое содержание

SPA-подходы.

Домашние задания

Клиентский роутинг и мини-SPA

Цель

изучить подходы SPA и клиентский роутинг и закрепить управление состоянием на уровне приложения.

Тема 3

Особенности деплоя и сборки SPA (CI/CD, Vercel, Netlify)

Цель занятия

освоить сборку и деплой фронтенда.

Краткое содержание

CI/CD; Vercel; Netlify.

Тема 4

Build tools: Vite, Webpack, Rollup, Web Components

Цель занятия

изучить инструменты сборки.

Краткое содержание

Webpack; Vite; Rollup; Web Components.

Домашние задания

Настройка сборки проекта (Vite/Webpack/Rollup)

Цель

освоить современный сборщик (Vite/Webpack/Rollup), настроить сборку HTML/CSS/JS и базовые инструменты качества.

Модуль 3. React и экосистема

Тема 1

Что такое React, JSX, настройка окружения (повторение ключевых основ)

Цель занятия

повторить основы React и JSX.

Краткое содержание

компоненты; JSX; окружение.

Тема 2

Hooks в React: углубляемся в детали useState, useEffect, useContext и кастомных хуков

Цель занятия

изучить продвинутые хуки React.

Краткое содержание

useState; useEffect; custom hooks.

Тема 3

Современные паттерны в React: (HOC, Render Props — исторический контекст, фокус на кастомных хуках)

Цель занятия

начать использовать паттерны React-разработки.

Краткое содержание

HOC; Render Props; кастомные хуки.

Домашние задания

Современные паттерны в React и кастомные хуки

Цель

начать использовать паттерны разработки в React и выносить логику в переиспользуемые кастомные хуки.

Тема 4

TypeScript в React: типизация компонентов, пропсов и хуков

Цель занятия

изучить TypeScript в React.

Краткое содержание

компоненты и хуки на TS.

Тема 5

Состояние приложения в React: Redux Toolkit, Zustand, React Query

Цель занятия

научиться управлять состоянием React-приложений.

Краткое содержание

Redux Toolkit; Zustand; React Query.

Тема 6

React Router, ленивая загрузка компонентов, Suspense, оптимизация React-приложения

Цель занятия

освоить роутинг и оптимизацию.

Краткое содержание

React Router; Suspense; Lazy loading.

Домашние задания

React Router, ленивая загрузка, Suspense

Цель

освоить маршрутизацию в React, ленивую загрузку компонентов и глобальное состояние маршрутизации.

Тема 7

Тестирование React-приложений: React Testing Library, Jest (упоминание E2E-тестирования)

Цель занятия

изучить тестирование React-компонентов.

Краткое содержание

тестирование; React Testing Library; Jest.

Домашние задания

Тестирование React-компонентов (React Testing Library, Jest)

Цель

научиться писать тесты для React-компонентов с использованием React Testing Library и Jest.

Тема 8

Консультация по ДЗ

Цель занятия

получить ответы на вопросы по ДЗ и по курсу.

Краткое содержание

затруднения при выполнении ДЗ; вопросы по программе.

Модуль 4. Vue и экосистема

Тема 1

Основы Vue3: компоненты, реактивность

Цель занятия

изучить Vue-компоненты.

Краткое содержание

компоненты Vue3; реактивность.

Тема 2

Управление компонентами во Vue: Composition API

Цель занятия

изучить Composition API.

Краткое содержание

Composition API; компоненты.

Домашние задания

Vue 3 Composition API — структура приложения

Цель

освоить Composition API и организовать структуру приложения на основе компонентов и композиционных функций.

Тема 3

Состояние приложения во Vue: Pinia (Vue-эк-исторический контекст)

Цель занятия

освоить состояние с Pinia.

Краткое содержание

Pinia; состояние Vue-приложений.

Тема 4

Routing и Advanced Vue: динамические маршруты, Nuxt (SSR)

Цель занятия

изучить продвинутый роутинг Vue.

Краткое содержание

vue-router; Nuxt; SSR.

Домашние задания

Vue Router и Nuxt (SSR)

Цель

освоить продвинутый роутинг во Vue: динамические маршруты, параметризованные пути, а также базовый SSR с Nuxt.

Тема 5

Консультация по ДЗ

Цель занятия

получить ответы на вопросы по ДЗ и по курсу.

Краткое содержание

затруднения при выполнении ДЗ; вопросы по программе.

Модуль 5. Архитектура и SSR

Тема 1

Архитектура фронтенд-приложений: микрофронтенды, Monogero (Nx, Turbogero), принципы SOLID

Цель занятия

изучить микрофронтенды, Monogero и SOLID.

Краткое содержание

микрофронтенды; Monogero (Nx, Turbogero); принципы SOLID.

Тема 2

Современные рендеринг-фреймворки и SSR: Next.js, Nuxt 3, SSG/ISR

Цель занятия

освоить серверный рендеринг (SSR).

Краткое содержание

Next.js; Nuxt 3; SSG/ISR.

Домашние задания

Минимальное SSR-приложение (Next.js/Nuxt 3)

Цель

освоить серверный рендеринг; собрать минимальное SSR-приложение на Next.js или Nuxt 3.

Модуль 6. Backend, серверные технологии и контейнеризация

Тема 1

Разработка собственного API

Цель занятия

создать собственный API.

Краткое содержание

разработка API; CRUD.

Тема 2

REST, RPC и сетевые запросы

Цель занятия

освоить сетевые запросы и протоколы.

Краткое содержание

REST; RPC; CRUD; сетевые запросы.

Домашние задания

REST API с CRUD (пример: «Клон LeetCode»)

Цель

освоить разработку REST API с CRUD-операциями.

Тема 3

Введение в Node.js и NPM, работа с браузером

Цель занятия

освоить Node.js и NPM.

Краткое содержание

Node.js; NPM; браузерные инструменты.

Тема 4

Под капотом Node.js: Event Loop, Timers, модули

Цель занятия

изучить внутреннюю работу Node.js.

Краткое содержание

Event Loop; Timers; модули Node.js.

Тема 5

Продвинутая серверная разработка на Nest.js

Цель занятия

освоить Nest.js.

Краткое содержание

фреймворк Nest.js; модули; контроллеры.

Тема 6

Базы данных и ORM: PostgreSQL, TypeORM или Prisma в связке с Nest.js

Цель занятия

освоить работу с базами данных и ORM.

Краткое содержание

PostgreSQL; TypeORM/Prisma.

Домашние задания

База данных и ORM (PostgreSQL + TypeORM/Prisma)

Цель

подключить базу данных PostgreSQL к приложению на Nest.js и настроить ORM с моделями и миграциями.

Тема 7

Контейнеризация (Docker) и интеграция с CI/CD

Цель занятия

освоить контейнеризацию и CI/CD.

Краткое содержание

Docker; CI/CD.

Модуль 7. Проектная работа

Тема 1

Вводное занятие по проектной работе. Обзор пройденных фреймворков и технологий

Цель занятия

выбрать и обсудить тему проектной работы; спланировать работу над проектом; ознакомиться с регламентом работы над проектом.

Краткое содержание

правила работы над проектом и специфика проведения итоговой защиты; требования к результату проекта и итоговой документации.

Домашние задания

Проектная работа

Цель

определить тему проектной работы, сформулировать задачу и подготовить план реализации с учётом регламента курса и требований к результату/документации и после согласования темы — реализовать MVP и сдать готовый проект.

Тема 2

Карьерная консультация

Цель занятия

подготовиться к собеседованиям.

Краткое содержание

портфолио; рекомендации по работе с вакансиями; тестовые задания; готовые компании; вопросы по проекту; софты; работа с резюме.